Open Civic Maintenance Layer

A Defensive Publication

Author: courter_baked

April 2026

Authorship Disclosure

The inventive concepts described in this document originated with and were developed by the author (courter_baked). This document was drafted with the assistance of an AI language model acting as a writing and technical articulation tool under the author's direction. All substantive technical and design decisions, including the core architecture, process sequence, and novel combinations described herein, are attributable to the human author.

Purpose of This Document

This document constitutes a defensive publication. Its purpose is to establish prior art for the systems and methods described herein, such that no party may subsequently patent these concepts and restrict their public availability. The author explicitly intends these ideas to remain open and freely usable by anyone.

System Overview

This document describes an open, robot-agnostic infrastructure for civic maintenance tasks — initially scoped to the detection and remediation of dog waste in public spaces, but generalized to support any camera-detectable ground-level civic maintenance task. The system is composed of three primary components:

- A computer vision backend that detects maintenance events via fixed cameras
- A geo-tagged job queue that logs events and dispatches them to robotic agents
- A novel camera calibration method using a standard smartphone as a self-describing optical calibration target

A supplementary system is also described: a fallback navigation system combining QR zone markers and camera-guided positioning for robots operating without GPS.

System 1: Smartphone Display CV Calibration

Problem Statement

Fixed outdoor cameras must be calibrated to translate pixel-space coordinates into real-world GPS coordinates. Traditional methods require specialized equipment, manual surveying, or network pairing between devices. This system eliminates all of those dependencies.

Method

During camera installation, the installer uses a standard smartphone running a lightweight application. The application displays a composite calibration screen visible to the camera, containing the following information rendered as machine-readable visual elements:

- A fiducial marker pattern (such as an ArUco or AprilTag marker) for pose estimation
- The smartphone's current GPS coordinates displayed in a large, high-contrast, OCR-readable font
- A GPS timestamp indicating the acquisition time and accuracy of the coordinate fix
- The smartphone's current gyroscope data (pitch, roll, yaw) displayed in the same readable format

The installer holds the phone screen toward the camera for a brief period — typically one to two seconds. The camera backend captures this frame and performs the following operations in a single pass:

- Detects the fiducial marker and solves for the camera's orientation (pitch, yaw, tilt) relative to the marker
- Reads the GPS coordinates optically via OCR or structured visual parsing
- Reads the GPS timestamp to assess fix quality
- Cross-validates the camera orientation solved from the fiducial pattern against the gyroscope data displayed on the screen

This single captured frame provides the camera backend with everything required to anchor itself in global coordinate space: its real-world GPS position (from the phone's location while adjacent to the camera mount), its orientation relative to the ground plane, and a confidence metric derived from the cross-validation of two independent orientation measurements.

Pixel-to-GPS Projection

Once calibrated, the camera backend uses the known mounting position, height, orientation, and field of view to project any detected pixel coordinate onto the ground plane and compute a real-world GPS coordinate. This allows detection events — such as a dog waste deposit — to be logged with a GPS coordinate derived entirely from the camera's field of view, without GPS hardware on the camera itself.

Key Novel Aspects

- The smartphone screen serves simultaneously as a GPS data carrier, a gyroscopic data carrier, and a fiducial calibration target — all read optically by the camera being calibrated
- No network pairing, Bluetooth, or API communication between the phone and the camera system is required
- No specialized hardware beyond a standard smartphone is needed for installation
- The calibration is self-validating via cross-reference of fiducial-derived orientation against gyroscope data displayed on screen
- Any camera capable of streaming video to a backend is compatible — no smart camera hardware required

Hardware Independence

Because all calibration intelligence resides in the backend software, the physical camera hardware is entirely commodity. Any IP camera, USB camera, or repurposed smartphone capable of streaming video is a valid node. This dramatically reduces deployment cost and enables community-driven infrastructure expansion without engineering expertise.

System 2: Fallback Navigation and Camera-Guided Positioning

Purpose

While GPS is the primary navigation method for robotic agents in this system, a QR-code-based fallback system is provided for robots operating without GPS, in GPS-denied environments (such as urban canyons or covered areas), or for hobbyist and repaired robots where GPS hardware is unavailable or non-functional.

Marker Design

Physical QR code markers are mounted at fixed, known locations within a camera's coverage zone — on lampposts, fence posts, walls, or other stable structures. Each QR code encodes a unique zone or marker identifier. The markers do not need to encode GPS coordinates directly; instead, the backend database associates each marker ID with its GPS position.

The camera backend, which can observe both the markers and any robots in its field of view, populates and maintains this association. When a camera is first deployed and calibrated (via System 1), it can automatically compute the GPS coordinates of any QR markers visible in its field of view using the same pixel-to-GPS projection described above. This means the physical QR markers are self-registering — no manual coordinate entry is required.

Robot Navigation Using Markers

A robot without GPS navigates by scanning QR markers as it moves through a zone. Upon reading a marker, the robot queries the backend for the associated GPS coordinate and its current job target, then receives movement vectors to navigate from its current known position to the target. The camera can also actively guide the robot by observing both the marker it last scanned and the target location, computing and transmitting real-time movement vectors.

Dual-Use Markers

For GPS-equipped robots, QR markers serve as drift correction anchors. The robot scans a marker, receives its known GPS coordinate, and reconciles this against its onboard GPS reading to correct accumulated navigation error. The same physical infrastructure thus serves both GPS and non-GPS robots without modification.

Camera-Guided Navigation Mode

In zones where the camera has full visibility of both the robot and the target location, the backend can act as an active navigation controller. The robot is identified visually (via a marker or distinctive feature on its chassis), and the camera backend computes and transmits movement vectors in real time. This mode requires only that the robot can receive and execute simple directional instructions, enabling very low-cost robotic hardware to participate in the system.

Closing Statement

The systems described in this document are released into the public domain as prior art. The author encourages open implementation, extension, and improvement of these concepts by any party. No permission is required to implement, publish, or build upon these ideas.

The author's intent is that when capable humanoid and wheeled robotic platforms become widely available, the infrastructure to deploy them for civic maintenance tasks will already exist as a mature, open, patent-free standard accessible to municipalities, communities, hobbyists, and commercial operators alike.